



DATABASE - MySQL

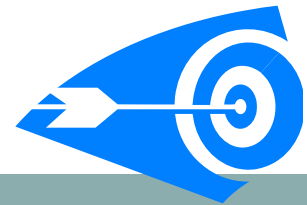


Muhammad Zen S. Hadi, ST. MSc.

Objectives

Tujuan:

Memahami perintah-perintah dasar DDL dan DML



Contents



- **Database**
 - DBMS
 - Relational model
- **SQL**
 - MySQL
 - MySQL yang support ke relational model yaitu innodb.
 - Standar tabel yang di MySQL adalah MyISAM.

Database



- **Database**
 - a collection of data
 - Entity, relationship
- **DBMS**
 - Database management system

Constraints



- **NOT NULL**
Tidak boleh berisi NULL (kosong).
- **UNIQUE**
Satu data dengan data yang lainnya tidak boleh sama.
- **PRIMARY KEY**
- **FOREIGN KEY**
Sebagai relasi antara 2 tabel.

START A MYSQL CLIENT



- **Connect**

- **mysql [-h host] [-u user] [-p[password]] [dbname]**

```
>mysql -u root -p
```

```
Enter password: *****
```

```
Welcome to the MySQL monitor.  Commands end with ;  
or \g.
```

```
Your MySQL connection id is 14 to server version:  
3.23.34a
```

```
Type 'help;' or '\h' for help. Type '\c' to clear  
the buffer
```

```
mysql>
```

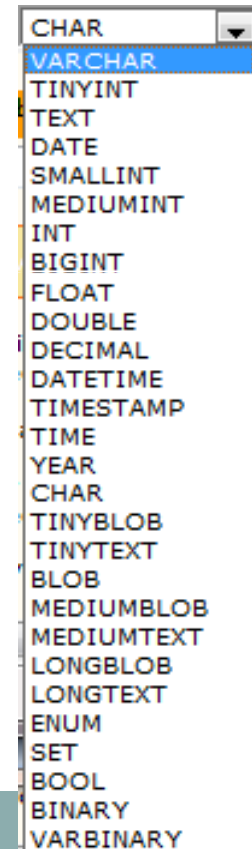
- Keluar dengan perintah quit atau exit.
- Setting ada di config.inc.php (untuk merubah user dan password)

Tipe data di MySQL



- Dalam bahasa SQL pada umumnya informasi tersimpan dalam tabel-tabel yang secara logik merupakan struktur dua dimensi terdiri dari baris (*row* atau *record*) dan kolom (*column* atau *field*). Sedangkan dalam sebuah *database* dapat terdiri dari beberapa *table*.
- Beberapa tipe data dalam MySQL yang sering dipakai:

Tipe data	Keterangan
INT(M) [UNSIGNED]	Angka -2147483648 s/d 2147483647
FLOAT(M,D)	Angka pecahan
DATE	Tanggal Format : YYYY-MM-DD
DATETIME	Tanggal dan Waktu Format : YYYY-MM-DD HH:MM:SS
CHAR(M)	String dengan panjang tetap sesuai dengan yang ditentukan. Panjangnya 1-255 karakter
VARCHAR(M)	String dengan panjang yang berubah-ubah sesuai dengan yang disimpan saat itu. Panjangnya 1 – 255 karakter
BLOB	Teks dengan panjang maksimum 65535 karakter
LOB	Teks dengan panjang maksimum 4294967295 karakter



MEMBUAT DATABASE DAN TABLE



- Cara untuk membuat sebuah database baru adalah dengan perintah:
create database *namadatabase*;
Contoh: create database privatdb;
- Untuk membuka sebuah database dapat menggunakan perintah berikut ini:
use *namadatabase*;
Contoh: use privatdb;
- Perintah untuk membuat tabel baru adalah:
create table *namatabel*
(
struktur
);
- Untuk DDL, gunakan phpmyadmin agar lebih mudah.

MySQL



- **Data Definition Language (DDL)**
 - **CREATE/DROP DATABASE dbname;**
 - **SHOW DATABASES;**
 - **USE dbname;**

 - **CREATE TABLE table_name (field_name type,..., constraints,..);**
 - **SHOW TABLES;**
 - **SHOW COLUMNS FROM table_name;**
 - **DROP TABLE table_name;**

MySQL



- **Data Manipulation Language (DML)**
 - **INSERT INTO table_name [(field_name,..)]
VALUES (value,..);**
 - **DELETE FROM table_name
WHERE condition;**
 - **UPDATE table_name
SET field_name=value,..
[WHERE condition];**

MySQL



- **Data manipulation(2)**
 - **SELECT field_name [as field_name],..
FROM table_name
[WHERE condition]
[ORDER BY field_name];**
 - **=, <, >, AND, OR, NOT
(field_name LIKE “_%....”)**

MEMBUAT DATABASE DAN TABLE



Contoh:

- Misalkan kita ingin menyimpan data anggota yaitu: nomor, nama, email, alamat, kota. Sedangkan strukturnya seperti tabel dibawah ini:

Kolom/Field	Tipe data	Keterangan
nomor	int(6) not null primary key	angka dengan panjang maksimal 6, sebagai <i>primary key</i> , tidak boleh kosong
nama	char(40) not null	teks dengan panjang maksimal 40 karakter, tidak boleh kosong
email	char(255) not null	teks dengan panjang maksimal 255 karakter, tidak boleh kosong
alamat	char(80) not null	teks dengan panjang maksimal 80 karakter, tidak boleh kosong
kota	char(20) not null	teks dengan panjang maksimal 20 karakter, tidak boleh kosong

Data yang diinginkan



Nomor	Nama	Email	Alamat	Kota
100	Adi	adi@yahoo.com	Jl. Keputih 2A no 5	Surabaya
101	Arif	arif@hotmail.com	Jl. Gebang Lor 32D	Surabaya
102	Iqbal	iqbal@yahoo.com	Jl. Klampis 3	Surabaya
103	Setyo	setyo@mailcity.com	Jl. Madiun 5	Sidoarjo
104	Ilham	ilham@yahoo.com	Jl. Surabaya 9	Malang
105	Syamsudin	sam@yahoo.com	Jl. Pabean 11	Surabaya
106	Faruq	faruq@yahoo.com	Jl. Raya 5	Kediri
107	Hari	hari@yahoo.com	Jl. Raya 9	Banyuwangi

Menggunakan phpMyAdmin



- software yang digunakan untuk membuat dan maintenance database
- Kita dapat mengakses database MySQL dengan account kita di phpMyAdmin.
- Pastikan server MySQL dan program phpMySQL sudah berjalan.
- untuk membuka tampilan phpMyAdmin di komputer lokal melalui URL
<http://localhost/phpmyadmin/>

Masukkan login sebagai root atau sesuai dengan user yang sudah dibuat



The screenshot displays the phpMyAdmin 2.6.2 interface. At the top, it says "Welcome to phpMyAdmin 2.6.2" and "MySQL 4.1.11-Debian_4-log running on localhost as root@localhost". The interface is divided into two main sections: "MySQL" and "phpMyAdmin".

MySQL Section:

- Create new database**: Includes a text input field, a "Collation" dropdown menu, and a "Create" button.
- Show MySQL runtime information**
- Show MySQL system variables**
- Show processes**
- Character Sets and Collations**
- Storage Engines**
- Reload MySQL**
- Privileges**
- Binary log**
- Databases**
- Export**
- Change password**
- Log out**

phpMyAdmin Section:

- Language**: Set to "English (en-utf-8)".
- MySQL charset**: Set to "UTF-8 Unicode (utf8)".
- MySQL connection collation**: Set to "utf8_general_ci".
- Theme / Style**: Set to "Original".
- phpMyAdmin documentation**
- Show PHP information**
- Official phpMyAdmin Homepage**
- [\[ChangeLog\]](#) [\[CVS\]](#) [\[Lists\]](#)

Left Sidebar:

- Database: (Databases) ...
- Please select a database

Menentukan field dan type data



Server: localhost ▶ Database: data ▶ Table: anggota

Field	Type ?	Length/Values ¹	Collation	Attributes	Null	Default ²	Extra			
nomor	INT	6			not null			<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
nama	CHAR	40			not null			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
email	CHAR	255			not null			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
alamat	CHAR	80			not null			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
kota	CHAR	20			not null			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Contoh Tabel



	Field	Type	Collation	Attributes	Null	Default	Extra	Action						
<input type="checkbox"/>	<u>nomor</u>	int(6)			No									
<input type="checkbox"/>	nama	char(40)	latin1_general_ci		No									
<input type="checkbox"/>	email	char(255)	latin1_general_ci		No									
<input type="checkbox"/>	alamat	char(80)	latin1_general_ci		No									
<input type="checkbox"/>	kota	char(20)	latin1_general_ci		No									

Check All / Uncheck All *With selected:*

Print view
 Relation view
 Propose table structure

Add field(s)
 At End of Table
 At Beginning of Table
 After

Indexes:

Keyname	Type	Cardinality	Action	Field
PRIMARY	PRIMARY	0		nomor

Create an index on columns

Space usage

Type	Usage
Data	0 Bytes
Index	0 Bytes
Total	0 Bytes

Row Statistics

Statements	Value

Create Tabel dengan phpmyadmin



Server: localhost Database: data

Structure **SQL** Search Query Export Import Operations Privileges Drop

Run SQL query/queries on database data: ?

```
create table anggota(  
  nomor int(6) not null primary key,  
  nama char(40) not null,  
  email char(255) not null,  
  alamat char(80) not null,  
  kota char(20) not null);
```

Bookmark this SQL query: Let every user access this bookmark Replace existing bookmark of same name

Show this query here again

MEMBUAT DATABASE DAN TABLE

- Perintah MySQL untuk membuat tabel seperti diatas adalah:

```
create table anggota(  
  nomor int(6) not null primary key,  
  nama char(40) not null,  
  email char(255) not null,  
  alamat char(80) not null,  
  kota char(20) not null);
```

- Untuk memasukkan sebuah baris (record) kedalam tabel MySQL adalah sebagai berikut:

```
insert into namatabel values(kolom1, kolom2,  
  kolom3,...);
```

Contoh:

```
insert into anggota values  
(106, 'faruq', 'faruq@yahoo.com', 'Jl. Raya 152', 'Kediri');
```

MENGHAPUS RECORD



- Untuk menghapus suatu record dengan kriteria tertentu digunakan perintah sebagai berikut:
`delete from namatabel where kriteria;`
- Contoh: Menghapus record dari tabel anggota yang bernomor '3'
`delete from anggota where nomor=3;`
- **DELETE FROM anggota;**

Will delete ALL anggota records!

MEMODIFIKASI RECORD



- Untuk memodifikasi (merubah) isi record tertentu adalah dengan menggunakan perintah sebagai berikut:

```
update namatabel set kolom1=nilaibaru1,  
kolom2=nilaibaru2 ... where kriteria;
```

- Contoh:Merubah e-mail dari anggota yang bernomor 12 menjadi 'supri@yahoo.com' dalam tabel anggota.

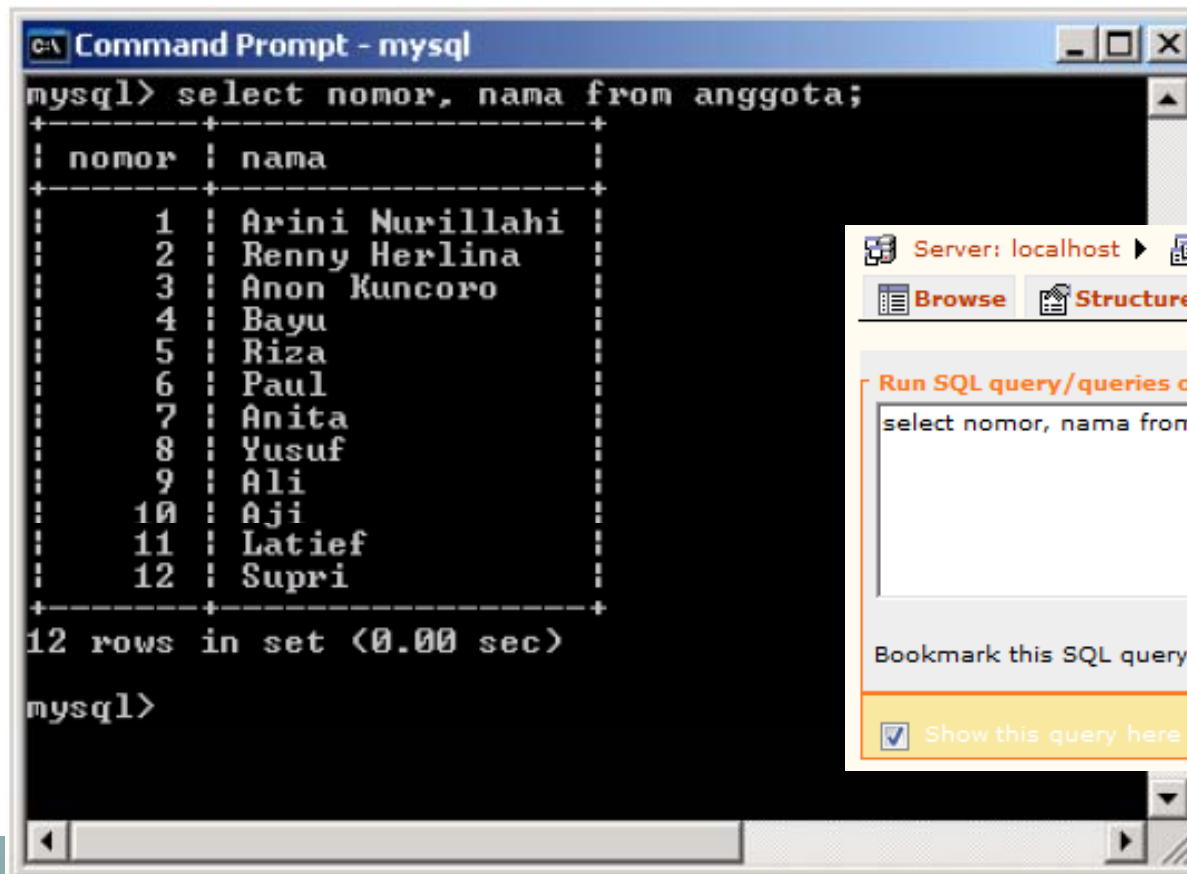
```
update anggota set email='supri@yahoo.com' where  
nomor=12;
```

- Untuk merubah kolom kota menjadi 'surabaya' semuanya dalam tabel anggota

```
update anggota set kota='surabaya';
```

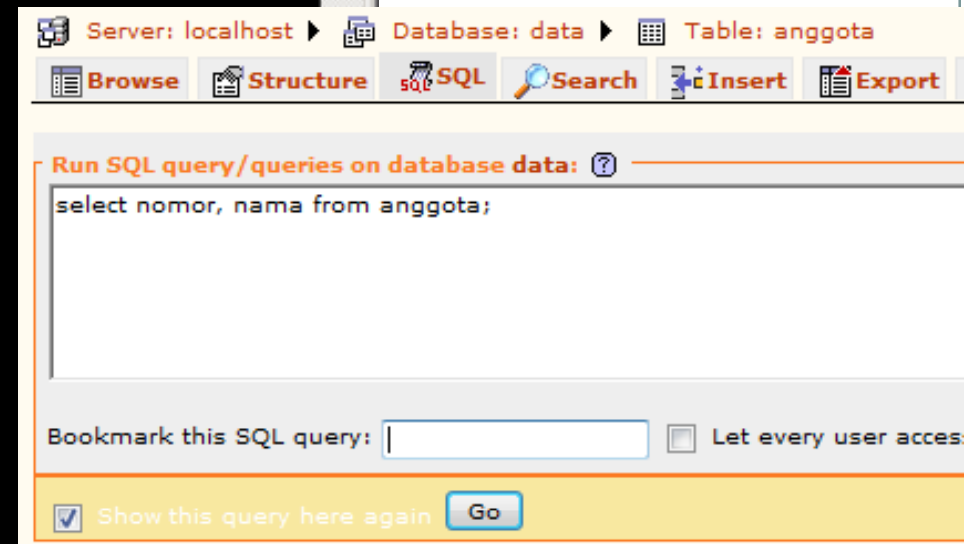
MENAMPILKAN ISI TABLE

- Isi tabel dapat ditampilkan dengan menggunakan perintah SELECT, cara penulisan perintah SELECT adalah:
`select nm_kolom from namatable;`



```
Command Prompt - mysql
mysql> select nomor, nama from anggota;
+----+-----+
| nomor | nama          |
+----+-----+
| 1     | Arini Nurillahi |
| 2     | Renny Herlina  |
| 3     | Anon Kuncoro   |
| 4     | Bayu           |
| 5     | Riza           |
| 6     | Paul           |
| 7     | Anita          |
| 8     | Yusuf          |
| 9     | Ali            |
| 10    | Aji            |
| 11    | Latief         |
| 12    | Supri          |
+----+-----+
12 rows in set (0.00 sec)

mysql>
```



Server: localhost Database: data Table: anggota

Browse Structure SQL Search Insert Export

Run SQL query/queries on database data: ?

```
select nomor, nama from anggota;
```

Bookmark this SQL query: Let every user access

Show this query here again

Konsep Dasar SQL Query



- Bentuk SQL Query umum:

```
SELECT [DISTINCT] < attribute-list >  
FROM < table-list >  
WHERE < condition >
```

- **Attribute- list:** adalah daftar nama atribut yang berada dalam *table-list* dan nilainya didapatkan melalui query.
- **Table- list:** adalah daftar relasi yang memiliki nama (dengan domain variabel pada tiap nama yang diberikan) untuk memproses query.
- **Condition:** adalah statemen perbandingan dalam SQL Query yang mengkombinasikan operator perbandingan AND, OR dan NOT.
- Sedangkan **DISTINCT** adalah keyword yang bersifat optional (boleh ditulis, boleh tidak) yang mengindikasikan suatu hasil query yang tidak memiliki duplikat. Secara default, didapatkan duplikasi pada hasil query (tanpa distinct).

Contoh Aplikasi (Google)



database



Telusuri

Sekitar 824,000,000 hasil (0.07 detik)

[Penelusuran lanjutan](#)

Semua

Gambar

Berita

Lainnya

Surabaya

[Ubah lokasi](#)

Web

[Laman dari Indonesia](#)

[Laman berbahasa asing yang telah diterjemahkan](#)

[Sebarang waktu](#)

[Basis data - Wikipedia bahasa Indonesia, ensiklopedia bebas](#) 🔍

Basis data (bahasa Inggris: **database**), atau sering pula dieja basisdata, adalah kumpulan informasi yang disimpan di dalam komputer secara sistematis ...

id.wikipedia.org/wiki/Basis_data - [Tembolok](#) - [Mirip](#)

[Database - Wikipedia, the free encyclopedia](#) 🔍 - [[Terjemahkan laman ini](#)]

A **database** is a system intended to organize, store, and retrieve large ...

en.wikipedia.org/wiki/Database - [Tembolok](#) - [Mirip](#)

[Tampilkan hasil lainnya dari wikipedia.org](#)

[Database : IlmuKomputer.Com](#) 🔍

Taufik Adi Sanjaya Website penulis : <http://pembuatwebdanblog.web.id> Dalam tulisan ini akan dijelaskan secara ringkas mengenai cara pengaksesan **database** ...

ilmukomputer.org/category/database/ - [Tembolok](#) - [Mirip](#)

[Database.com](#) 🔍 - [[Terjemahkan laman ini](#)]

Database.com – Built for the Cloud. The latest articles, white papers, toolkits, blogs and documentation about **Database.com**.

SQL – Menghilangkan Data Row Duplikat



```
SELECT department_id  
FROM employees;
```

DEPARTMENT_ID
90
90
90
60
60
60
50
50
50

...

20 rows selected.

```
SELECT DISTINCT department_id  
FROM employees;
```

DEPARTMENT_ID
10
20
50
60
80
90
110

8 rows selected.

S Q L – Operasi Pembanding



Operator	Meaning
=	Equal to
>	Greater than
>=	Greater than or equal to
<	Less than
<=	Less than or equal to
<>	Not equal to

```
SELECT last_name, salary
FROM employees
WHERE salary <= 3000;
```

LAST_NAME	SALARY
Matos	2600
Vargas	2500

S Q L – Operasi Pembanding

Other Comparison Conditions

Operator	Meaning
BETWEEN ...AND...	Between two values (inclusive),
IN(set)	Match any of a list of values
LIKE	Match a character pattern
IS NULL	Is a null value

Using the BETWEEN Condition

```
SELECT last_name, salary  
FROM employees  
WHERE salary BETWEEN 2500 AND 3500;
```

Lower limit

Upper limit

LAST_NAME	SALARY
Rajs	3500
Davies	3100
Matos	2600
Vargas	2500

S Q L – Operasi Pembanding

Using the IN Condition

```
SELECT employee_id, last_name, salary, manager_id  
FROM employees  
WHERE manager_id IN (100, 101, 201);
```

EMPLOYEE_ID	LAST_NAME	SALARY	MANAGER_ID
202	Fay	6000	201
200	Whalen	4400	101
205	Higgins	12000	101
101	Kochhar	17000	100
102	De Haan	17000	100
124	Mourgos	5800	100
149	Zlotkey	10500	100
201	Hartstein	13000	100

8 rows selected.

S Q L – Operasi Pembanding



Using the **LIKE** Condition

- Gunakan kondisi **LIKE** untuk melakukan pencarian sebagian nilai string.
- Kondisi pencarian dapat menggunakan simbol karakter berikut:
 - **%** : menunjukkan nol/kosong atau sembarang **beberapa** karakter.
 - **_** : menunjukkan sembarang **1** karakter.

SQL – Operasi Pembanding

```
SELECT first_name  
FROM employees  
WHERE first_name LIKE 'S%';
```

FIRST_NAME
Steven
Shelli
Sigal
Shanta
Steven
Stephen
Sarath
Shelley

8 rows selected.

```
SELECT last_name  
FROM employees  
WHERE last_name LIKE '_o%';
```

LAST_NAME
Kochhar
Lorentz
Popp
Rogers
Doran
Fox
Johnson
Jones

8 rows selected.

S Q L – Operasi Pembanding



Logical Conditions

Operator	Arti
AND	Returns TRUE , jika kedua kondisi adalah True
OR	Returns TRUE , jika salah satu kondisi adalah True
NOT	Returns TRUE , jika kondisi tersebut adalah False

S Q L – Operasi Pembanding

Using the AND Operator

```
SELECT employee_id, last_name, job_id, salary
FROM employees
WHERE salary >=10000
AND job_id LIKE '%MAN%';
```

EMPLOYEE_ID	LAST_NAME	JOB_ID	SALARY
149	Zlotkey	SA_MAN	10500
201	Hartstein	MK_MAN	13000

S Q L – Operasi Pembanding

Using the OR Operator

```
SELECT employee_id, last_name, job_id, salary
FROM employees
WHERE salary >= 10000
OR job_id LIKE '%MAN%';
```

EMPLOYEE_ID	LAST_NAME	JOB_ID	SALARY
100	King	AD_PRES	24000
101	Kochhar	AD_VP	17000
102	De Haan	AD_VP	17000
124	Mourgos	ST_MAN	5800
149	Zlotkey	SA_MAN	10500
174	Abel	SA_REP	11000
201	Hartstein	MK_MAN	13000
205	Higgins	AC_MGR	12000

8 rows selected.

SQL – JOIN



- Syntax Join SQL:

```
SELECT  table1.column, table2.column
FROM    table1, table2
WHERE   table1.column1 = table2.column2;
```

SQL – Join

- **Join** adalah bentuk kondisi join dimana nilai relasi yang terjadi antar dua atau lebih table (binary relation) adalah sama (terdapat hubungan antara Primary Key dan Foreign Key)

Contoh:

EMPLOYEES		DEPARTMENTS	
EMPLOYEE_ID	DEPARTMENT_ID	DEPARTMENT_ID	DEPARTMENT_NAME
200	10	10	Administration
201	20	20	Marketing
202	20	20	Marketing
124	50	50	Shipping
141	50	50	Shipping
142	50	50	Shipping
143	50	50	Shipping
144	50	50	Shipping
103	60	60	IT
104	60	60	IT
107	60	60	IT
149	80	80	Sales
174	80	80	Sales
176	80	80	Sales
...

Foreign key Primary key

SQL – Join

Retrieving Records with joins

```
SELECT employees.employee_id, employees.last_name,  
       employees.department_id, departments.department_id,  
       departments.location_id  
FROM   employees, departments  
WHERE  employees.department_id = departments.department_id;
```

EMPLOYEE_ID	LAST_NAME	DEPARTMENT_ID	DEPARTMENT_ID	LOCATION_ID
200	Whalen	10	10	1700
201	Hartstein	20	20	1800
202	Fay	20	20	1800
124	Mourgos	50	50	1500
141	Rajs	50	50	1500
142	Davies	50	50	1500
143	Matos	50	50	1500
144	Vargas	50	50	1500

■ ■ ■

19 rows selected.

SQL – Join

Kondisi Join dengan Operator AND

```
SELECT last_name, employees.department_id,  
       department_name  
FROM   employees, departments  
WHERE  employees.department_id = departments.department_id  
AND    last_name = 'Matos';
```

EMPLOYEES

LAST_NAME	DEPARTMENT_ID
Whalen	10
Hartstein	20
Fay	20
Mourgos	50
Rajs	50
Davies	50
Matos	50
Vargas	50
Hunold	60
Ernst	60

...

DEPARTMENTS

DEPARTMENT_ID	DEPARTMENT_NAME
10	Administration
20	Marketing
20	Marketing
50	Shipping
50	Shipping
50	Shipping
50	Shipping
50	Shipping
60	IT
60	IT

...

SQL – Join

Kondisi Join Menggunakan Table Alias

- Menyederhanakan queries dengan menggunakan table alias.
- Meningkatkan performance.

```
SELECT e.employee_id, e.last_name, e.department_id,  
       d.department_id, d.location_id  
FROM   employees e , departments d  
WHERE  e.department_id = d.department_id;
```

SQL – Join

Kondisi Join Menggunakan Table Alias

```
SELECT e.last_name, d.department_name, l.city
FROM   employees e, departments d, locations l
WHERE  e.department_id = d.department_id
AND    d.location_id = l.location_id;
```

EMPLOYEES

LAST_NAME	DEPARTMENT_ID
King	90
Kochhar	90
De Haan	90
Hunold	60
Ernst	60
Lorentz	60
Mourgos	50
Rajs	50
Davies	50
Matos	50
Vargas	50
Zlotkey	80
Abel	80
Taylor	80

DEPARTMENTS

DEPARTMENT_ID	LOCATION_ID
10	1700
20	1800
50	1500
60	1400
80	2500
90	1700
110	1700
190	1700

8 rows selected.

LOCATIONS








LOCATION_ID	CITY
1400	Southlake
1500	South San Francisco
1700	Seattle
1800	Toronto
2500	Oxford

20 rows selected.

Data Field dengan spasi



	Field	Type	Collation	Attributes	Null	De
<input type="checkbox"/>	tukar uang	varchar(20)	latin1_general_ci		No	
<input type="checkbox"/>	nilai	int(10)			No	

↑ Check All / Uncheck All With selected:       

- Jika ingin perintah select :

```
select * from uang where `tukar uang` = 'euro';
```

```
select * from uang where `tukar uang` = "euro";
```

Perhatikan tanda pada tukar uang beda dengan euro

Latihan Soal

- Diketahui tabel anggota berikut ini:

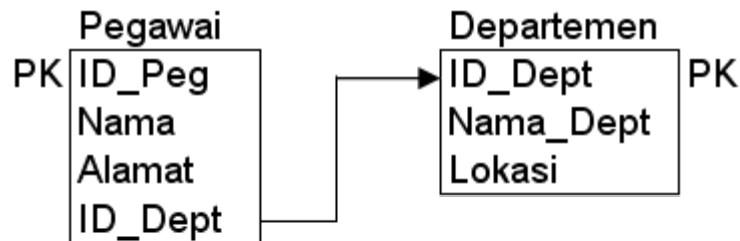
ID	Nama	Alamat	Gaji
100	Arif	Surabaya	10000
101	Andi	Jakarta	14000
102	Burhan	Malang	12000
103	Fikri	Madiun	15000
104	Fariz	Malang	17000
105	Sigit	Surabaya	20000
106	Ifan	Kediri	16000
107	Hanif	Yogyakarta	12000
108	Zakiuddin	Surabaya	21000

- Buatlah perintah select
 - a. Tampilkan data ID dan nama yang berdomisili di Surabaya.
 - b. Tampilkan nama yang gajinya diatas 15000
 - c. Tampilkan nama yang berawalan huruf F
 - d. Tampilkan nama yang berakhiran f

Latihan Soal



- Diketahui relasi tabel sbb:

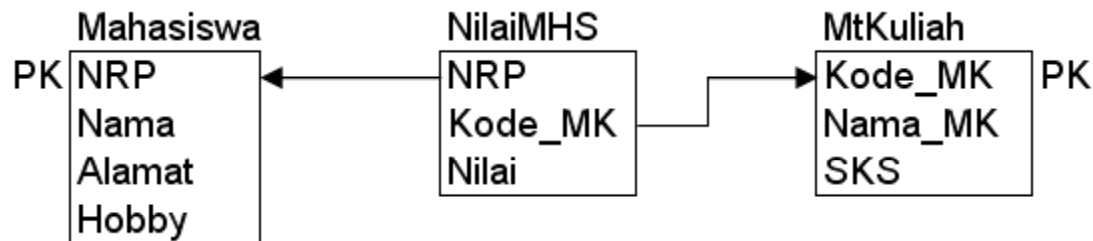


- Buat perintah untuk :
 - a. Menambahkan data di tabel pegawai
 - b. Mengedit data di tabel departemen
 - c. Delete data di tabel pegawai yang beralamat di Surabaya
 - d. Tampilkan data Nama dan alamat pegawai dan nama departemen yang berdomisili di jakarta

Latihan Soal



- Diketahui relasi tabel sbb:



- Buat perintah untuk :
 - a. Menambahkan data di tabel mahasiswa, nilaiMHS dan MtKuliah
 - b. Mengedit data di tabel NilaiMHS
 - c. Delete data di tabel MtKuliah yang berkode VT001
 - d. Tampilkan data Nama dan alamat mahasiswa, nama mata kuliah, SKS dan nilainya.